**Rich Kiddo!**

You are from rich land. You own every note from 1, 2, 3, ….. ,∞. But you have **only 1 note** of each kind ( one $1, one $2, one $3…. ). You want to go to after-eid shopping at Rich Future Park aka RFP.

RFP is a huge N storied building. Every floor has N shops. As you are really a rich kiddo, you want to buy from every floor. But you decided to buy only 1 item from each floor. Now, Shamim wants to check if you can spend optimally. He asks you, in how many ways you can buy 1 item from every floor such that you spend as minimum as possible. Two ways are considered different if one of them has a different shop than other.

|  |  |  |
| --- | --- | --- |
| 2 | **1** | 3 |
| **3** | 2 | 2 |
| 5 | 4 | **2** |

A 3-storied building with 3 shop on each floor

For the given example, you can spend $6 ( 1 + 3 + 2 ) which is minimum among all other configurations and there is only 1 way to achieve that. Configuration (1 + 2 + 2) is invalid. Because you can spend $2 only once.

**INPUT**

The first line contains a single integer T (1 <= T <= 100) – The number of test cases.

The first line of each test case contains a single integer N (1 <= N <= 5) – Meaning RFP is N storied building.

Then N lines follow. Each line contains N integers (1 <= A[i][j] <= 1000). Each integer represents the price of a single item.

**OUTPUT**

For each test case output two space separated integers. Number of ways W and the minimum amount M that you will spend. If there is no valid way to buy, print “0 0” without quotes.

|  |  |
| --- | --- |
| **SAMPLE INPUT** | **SAMPLE OUTPUT** |
| 3  3  2 1 3  3 2 2  5 4 2  4  1 2 3 4  1 2 3 4  1 2 3 4  1 2 3 4  5  1 2 3 4 5  5 4 3 5 4  43 2 5 4 1  1 2 36 7 2  2 3 8 7 3 | 1 6  24 10  54 15 |

Tag: bruteforce